

How to Remove the First and Last Character from Text in Excel

Authored by
stats writer

February 20, 2026

RECOMMENDED CITATION

stats writer (2026). *How to Remove the First and Last Character from Text in Excel*. PSYCHOLOGICAL SCALES. Retrieved from <https://scales.arabpsychology.com/?p=131691>

The Importance of Data Transformation in Microsoft Excel

In the modern data-driven landscape, **Microsoft Excel** remains an indispensable tool for professionals across various industries. Whether you are a **data analyst**, an accountant, or a project manager, the ability to clean and format raw data is a foundational skill. Often, when importing data from external databases or **CSV** files, **string** values may contain unwanted formatting, such as leading and trailing quotation marks, brackets, or legacy system codes. Successfully removing these characters is essential for ensuring that your **data cleaning** process is thorough and that subsequent calculations or lookups function correctly without errors caused by stray characters.

Mastering **text manipulation** techniques allows users to transform messy, inconsistent datasets into streamlined, usable information. This process of **string manipulation** is not merely about aesthetics; it is about data integrity. For instance, if you are attempting to perform a **VLOOKUP** or a **XLOOKUP** on a list of IDs where some entries have extra characters, the formula will fail to find a match. By learning to strip the first and last characters from a **string**, you enhance your efficiency and reduce the risk of manual entry errors that often plague large-scale **spreadsheet** projects.

The flexibility of **Excel** functions provides multiple pathways to achieve this goal. While some users might prefer manual editing for small lists, professional workflows demand automated solutions. Utilizing a combination of built-in **functions** allows for dynamic updates; as the source data changes, the output reflects those changes instantly. This tutorial will explore the most efficient methods for removing the first and last characters from a **string**, focusing on the **MID** and **LEN** functions, while also touching upon alternative approaches such as **Power Query** for more complex scenarios.

The Mechanics of String Indexing and Manipulation

To understand how to remove specific characters from a **string**, one must first understand how **Microsoft Excel** perceives text. Each **string** is treated as a sequence of individual characters, each occupying a specific numerical position or index. For example, in the word "Excel", the character 'E' is at position 1, and 'l' is at position 5. When we talk about removing the first and last characters, we are essentially telling the software to extract a **substring** that begins at the second position and ends exactly one character before the final index.

This conceptual framework is vital when constructing a **formula**. Unlike some programming languages that use zero-based indexing, **Excel** uses one-based indexing. This means the first character is always at position 1. Therefore, to skip the first character, our starting point must always be 2. The challenge then becomes determining where to stop. Since different **cells** may contain strings of varying lengths, we cannot use a static number. This is where dynamic **logic**

becomes necessary, allowing the formula to adapt to the specific length of the text in each **cell**.

Furthermore, it is important to recognize that **whitespace** characters are counted just like letters or numbers. A trailing space at the end of a **string** will be treated as the "last character." If your goal is to remove a visible character like a parenthesis but there is a hidden space after it, a standard formula might only remove the space. Consequently, understanding the **string** structure and utilizing helper functions like **TRIM** or **CLEAN** can be a necessary precursor to more advanced **text manipulation** tasks in a professional **spreadsheet** environment.

Deep Dive into the MID Function Syntax

The **MID function** is the primary tool used for extracting a specific portion of text from the middle of a **string**. The syntax for this **function** is relatively straightforward: `MID(text, start_num, num_chars)`. The first argument, "text," refers to the **cell** containing the **string** you wish to modify. The second argument, "start_num," specifies the position of the first character you want to extract. For our specific purpose of removing the first character, this value will always be set to 2, effectively instructing **Excel** to ignore the character at the very beginning of the **string**.

The third argument, "num_chars," is where the complexity lies. This argument tells **Excel** how many characters to include in the output, starting from the "start_num." If we were to use a fixed number here, the formula would only work for strings of a specific length. However, in most real-world datasets, the length of the **string** varies from one row to the next. To make this argument dynamic, we must subtract the total number of unwanted characters from the total length of the **string**. In the case of removing both the first and the last character, we are removing exactly two characters in total.

By combining these elements, the **MID function** becomes a surgical tool for data extraction. It is part of a larger suite of **text functions** provided by **Microsoft Support**, which also includes **LEFT** and **RIGHT**. While **LEFT** extracts from the start and **RIGHT** from the end, **MID** offers the unique capability to define an internal range, making it the most efficient choice for stripping both ends of a **string** simultaneously without needing to nest multiple complex **functions**.

Leveraging the LEN Function for Dynamic Text Length

The **LEN function** is a simple yet powerful utility that returns the total number of characters in a **string**. In the context of our **formula**, **LEN** serves as the engine that allows the **MID function** to handle variable text lengths. Without **LEN**, we would have no way of telling **Excel** exactly where the **string** ends. By calculating the total length and then subtracting two, we create a mathematical instruction that says: "Take everything starting from the second character, but stop before you reach the very last one."

For example, if a **cell** contains the **string** "", the **LEN function** would return a value of 7. To remove the brackets, we start at position 2 and want to extract the "12345" part, which is 5 characters long. Using the math $7 - 2 = 5$, our formula identifies exactly how many characters to pull. If the next **cell** contains "", **LEN** returns 5, and the math $5 - 2 = 3$ correctly extracts the three digits. This level of automation is what makes **spreadsheet** software so powerful for **data cleaning** at scale.

It is worth noting that the **LEN function** is sensitive to all characters, including non-printable ones and **ASCII** symbols. When working with data exported from web applications or legacy **databases**, you might encounter hidden characters that cause **LEN** to return a higher value than expected. In such cases, the result of your **string manipulation** might look incorrect. To prevent this, many professionals wrap their target **cell** in a **TRIM** function within the **LEN** calculation to ensure that leading or trailing **whitespace** does not interfere with the character count.

Step-by-Step Implementation Guide

The process of removing the first and last character from a string in **Excel** involves utilizing the built-in functions of the software. This can be achieved by using the **LEFT** and **RIGHT** functions to extract the desired characters, and then combining them using the **CONCATENATE** function. This method allows for the removal of the first and last characters, without affecting the rest of the string. Additionally, the use of the **SUBSTITUTE** function can also be employed to replace the first and last characters with a blank space, effectively removing them from the string. Overall, these functions provide a simple and efficient way to remove the first and last character from a string in **Excel**.

Excel: Remove First and Last Character from String

In many practical scenarios, you may find it necessary to eliminate both the initial and the concluding characters from a specific string within a spreadsheet.

To accomplish this efficiently, you can implement the following formula:

=MID(A2,2,LEN(A2)-2)

This specific formula is designed to strip the first and last character from the string located in cell A2.

For instance, if cell A2 contains the value AA4506, the application of this formula would result in the output A450.

The following detailed example demonstrates how to apply this logic in a real-world Excel environment.

Example: Remove First and Last Character from String in Excel

Imagine a scenario where we have a dataset containing a list of employee identification numbers as shown below:

	A	B	C	D	E
1	Employee ID				
2	AA4506				
3	BV9002				
4	GH3484				
5	FH3400				
6	AA2003				
7	AB3044				
8	AC9003				
9	DV4599				
10	AR3040				
11					
12					
13					
14					
15					
16					
17					

In this case, our objective is to remove the leading and trailing characters from every employee ID in the list to isolate the core numeric value.

To achieve this, we enter the following string manipulation formula into cell B2:

=MID(A2,2,LEN(A2)-2)

After entering the formula, we can utilize the fill handle to drag the formula down through the remaining cells in column B, applying the logic to the entire dataset:

	A	B	C	D
1	Employee ID	First & Last Chacters Removed		
2	AA4506	A450		
3	BV9002	V900		
4	GH3484	H348		
5	FH3400	H340		
6	AA2003	A200		
7	AB3044	B304		
8	AC9003	C900		
9	DV4599	V459		
10	AR3040	R304		
11				
12				
13				
14				

As a result, Column B now displays the updated employee IDs from Column A, successfully excluding the first and last characters from each entry.

How This Formula Works

Let us break down the components of the formula used to remove the characters from cell A2:

=MID(A2,2,LEN(A2)-2)

The MID function in Excel is specifically built to extract a designated number of characters from any part of a string, based on a defined starting position and the

quantity of characters required for extraction.

The LEN function is a standard Excel tool used to calculate the exact string length within a given cell.

By nesting these functions, we instruct Excel to begin the extraction at position 2 and continue for a length equal to the total number of characters minus two.

This mathematical approach effectively isolates the central portion of the string, leaving out the unwanted characters at both ends.

The final output is a perfectly cleaned string that maintains the integrity of the original data while removing the peripheral noise.

Note: It is important to remember that blank spaces within the cell are treated as characters. To ensure accuracy, you might need to apply data cleaning techniques like removing spaces before running the extraction formula.

Summary of Related Excel Operations

The following resources and tutorials offer further

insight into common Excel operations and text manipulation strategies for advanced users:

Common Challenges: Spaces, Symbols, and Special Characters

While the MID and LEN combination is highly effective, users often encounter challenges when dealing with dirty data. One of the most common issues is the presence of whitespace. In Microsoft Excel, a space is considered a character just like any letter or digit. If your string has a trailing space that you cannot see, the LEN function will count it, and your formula will "remove" that space as the last character, leaving the actual character you wanted to remove still in place. To combat this, professionals often wrap their references in the TRIM function, which removes all leading and trailing spaces.

Another challenge involves non-printable characters that often hide in data imported from HTML sources or complex databases. These characters can interfere with the character count and the string manipulation logic. Using the CLEAN function in conjunction with your formula can help strip these invisible characters before the extraction takes place. This ensures that the "first"

and "last" characters identified by Excel are the actual visible characters you intend to target, leading to much more reliable results in your data cleaning workflows.

Finally, consider the case of very short strings. If a string has only one or two characters, subtracting two from the length will result in zero or a negative number, which will cause the MID function to return an empty string or an error. When working with unpredictable datasets, it is a best practice to wrap your formula in an IF or IFERROR statement. This allows you to handle edge cases gracefully, perhaps by returning the original string or a custom message if the text is too short to have both a first and last character removed.

Alternative Methods for Advanced Text Extraction

While the MID function is the most common solution, there are other ways to remove the first and last characters depending on your specific version of Excel or the complexity of your task. For instance, in Excel 365, the new TEXTBEFORE and TEXTAFTER functions can sometimes be leveraged for similar results, though they usually require a specific delimiter. Alternatively, using REPLACE can be a viable strategy: `=REPLACE(REPLACE(A2,`

`LEN(A2), 1, "") , 1, 1, ""`). This method works by replacing the last character with nothing and then doing the same for the first character.

For users dealing with massive datasets that require repetitive data cleaning, Power Query (also known as Get & Transform) is a superior alternative. Power Query provides a graphical user interface where you can split columns by number of characters or use the "Transform" tab to remove a specific number of starting and ending characters. This approach is often preferred for Big Data tasks because it records your steps and can be refreshed whenever new data is added to the source, providing a more robust and scalable logic than standard cell formulas.

Lastly, for those comfortable with VBA (Visual Basic for Applications) or Office Scripts, a custom User Defined Function (UDF) can be created to handle this task. This is particularly useful if you need to perform this specific string manipulation across many different workbooks and want to simplify the process for other users who may not be familiar with complex formula nesting. By creating a function like `=RemoveEnds(text)`, you can streamline

the user experience while maintaining the high level of precision required for professional data management.

ARABPSYCHOLOGY.COM